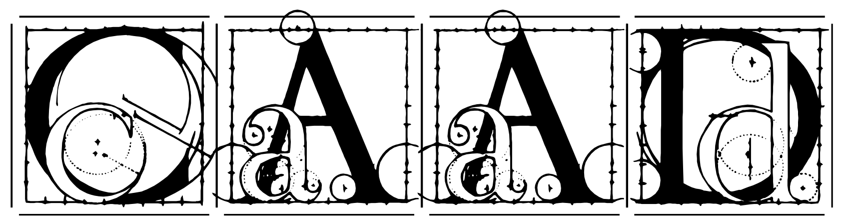


Boletín sobre Aventuras Conversacionales

Número 42 – Segunda Edad – Año 12

Junio-2000 http://pagina.de/caad

Inventario:

También puedes ver a:

- Javier San José (JSJ) (jsjlogin@jet.es)
- Roberto Ruisánchez (Deckard)
- Fernando Rodríguez
- Francisco Prieto (fprietog@retemail.es)
- José Baltasar García Pérez-Schofield (jbgarcia@uvigo.es)

La Inquisición

Si por fuera el edificio resulta impresionante, por dentro no lo es menos. Varios acólitos se ocupan de destruir metódicamente extraños artefactos mágicos. Al fondo te espera **El Gran Inquisidor**, se levanta cuando te ve llegar y dice...

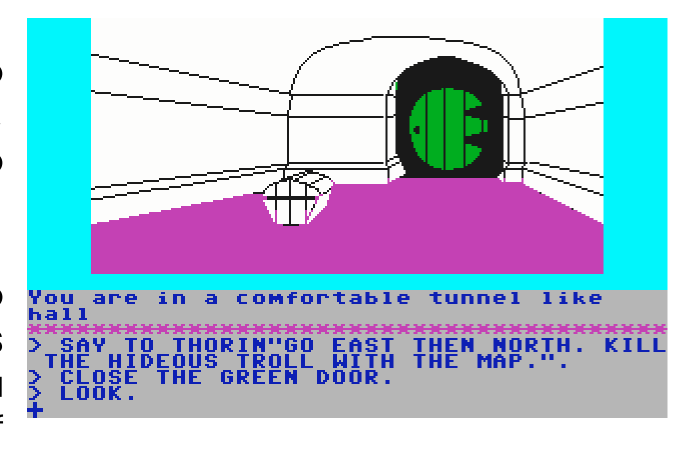
Con este número se abre una nueva etapa del decano fanzine CAAD. Una "segunda edad" que tiene como principal meta el conseguir una mayor periodicidad. No voy a decir que no vaya a haber retrasos ya que la experiencia acumulada a las espaldas demuestra que eso no siempre es posible. Lo que sí puedo decir es que se intentará que la periodicidad de este fanzine se mantenga mensual.

El Gran Inquisidor

La Cueva Colosal

Junto a un improvisado fuego se encuentra una semielfa que te mira divertida. "Hola viajero, mi nombre es Xzira. Pasa y ponte cómodo, aquí nada ni nadie puede dañarte.".

La femenina y sensual voz te sumerge en relatos de antiguas hazañas de pequeños hobbits que luchan contra enormes dragones...



"Aquel hobbit, Bilbo se llamaba, si que supo como enfrentarse al terrorífico Smaug. Tras leer el mapa que Gandalf le entregó (READ MAP) se dispuso a preparar lo que sería el viaje de su vida. Gandalf convenció a esos enanos de que necesitaban un ladrón en el grupo y Bilbo dejó su cómoda vida de hobbit para embarcarse en la aventura de su vida. El enano Thorin era receloso de las habilidades de este hobbit así que no le quitó ojo durante todo el viaje. Tras una larga caminata hacia el este (E, E) el grupo llegó al Claro de los Trolls. Dos enormes trolls esperaban en el claro. Bilbo demostró a todos su ingenio. Se dirigió sigilosamente hacia el norte y esperó a que amaneciese (N, WAIT). Con gran sorpresa todos volvieron al claro (S) para encontrar a los feroces trolls convertidos en estatuas de piedra. Bilbo recogió la llave y se dirigió a la cueva de los trolls donde encontró algunas cosas que les resultarían útiles a lo largo de su recién empezado viaje (N, UNLOCK DOOR, OPEN IT, N, GET ROPE AND SWORD).

La siguiente etapa del viaje (S, S, SE) les llevó derechos hasta Elrond, un elfo que les ayudó a descifrar el mapa de Gandalf y les proporcionó víveres (GIVE MAP TO ELROND, SAY TO ELRON "READ MAP", SAY TO ELROND "GIVE ME MAP"). Ahora estaba claro el siguiente paso gracias a la información del mapa (E, S, E, N, NW, N, SE, D, D, D, D, E, GET GOLDEN KEY, U, N, W, S, E, N). Después de esta larga caminata la compañía decidió parar para descansar y aclarar ideas. Mientras todos discutían (WAIT, **hasta que se abra la grieta**) sobre cuál era el siguiente paso a dar, una grieta se abrió en la

pared y Bilbo y Thorin fueron capturados por unos goblins. Ambos fueron a dar con sus huesos en una celda en el complejo de cavernas de los goblins. El ingenio de Bilbo empezó a funcionar de nuevo, como era habitual en este hobbit en situaciones comprometidas, y comenzó a cavar en una zona arenosa de la celda que parecía haber sido removida hace poco (**DIG**). Después de varios intentos (Thorin pasaba el rato sentado, cantando sobre oro y riquezas) fué a dar con una trampilla de madera cerrada. Presa de la frustración, el hobbit comenzó a asestar furiosos golpes contra la trampilla, una y otra vez (**SMASH TRAP DOOR**). Todo este esfuerzo no sirvió sino para desvelar un pequeño almacen donde Thorin encontró una curiosa llave. El enfado de Bilbo por no haber encontrado una salida fue tal que empezó a pasear por la celda mirando hacia el techo con tanta casualidad que en lo alto divisó una ventana, demasiado alta para alcanzarla. Demasiado alta para la escasa estatura de un hobbit pero Thorin parecía poder alcanzarla. Bilbo indicó a Thorin que abriese la ventana (estos goblins a veces eran tan estúpidos) y que le aupase y ambos salieron de la celda (**SAY THORIN "OPEN WINDOW", SAY THORIN "PICK ME UP", SAY THORIN "GO WINDOW"**).

Era sabido que los goblins construían cuevas de forma laberíntica y desordenada, así que nuestros amigos se pusieron en marcha en busca de la salida (**SW, D**). A lo lejos se oían los pasos de las patrullas goblins así que decidieron esperar hasta que pasasen de largo (**WAIT, hasta que aparezca un goblin**). Pasado el peligro siguieron caminando guiados por su instinto (**N, SE, E**) hasta llegar a una oscura cueva en la que habitaba un misterioso personaje. Gollum, un ser repugnante que no paraba de hablar de su precioso. Ambos no le prestaron la más mínima atención a este personaje al que parecían gustarle las adivinanzas. Bilbo encontró, por casualidad un pequeño anillo de oro que se guardó enseguida (**GET RING**) y prosiguieron en busca de la salida (**N, S, NW, E**). Una pequeña puerta era el último obstáculo hacia la luz del sol (**OPEN DOOR, U, CLOSE DOOR**). Ambos continuaron caminando, ya separados del



resto del grupo (E, E) hasta llegar a una casa. El humo de la chimenea y el olor a comida recién hecha les incitó a entrar. Curiosamente no había nadie en la casa. Bilbo encontró un poco de comida en un armario que procedió a engullir haciendo caso omiso a las prisas de Thorin (**OPEN CURTAIN, OPEN CUPBOARD, GET FOOD, EAT IT**).

De vuelta al camino siguieron en dirección norte y luego hacia el este (**NE, E, E**) hasta llegar a un tenebroso río demasiado caudaloso como para vadearlo. Bilbo volvió a sorprender al enano debido a su aguzada vista (**LOOK ACROSS RIVER**) que le permitió divisar un bote de remos al otro lado. Haciendo uso de la cuerda consiguió, con cierta habilidad y tras varios intentos, arrastrarlo hacia la orilla en la que se encontraban (**THROW ROPE ACROSS RIVER, PULL ROPE**). Como quiera que los enanos no son muy amigos del agua, Bilbo tuvo que insistir varias veces hasta que Thorin decidió, a regañadiente, subir al bote (**SAY TO THORIN "CLIMB INTO BOAT"**) y luego subió él (**CLIMB INTO BOAT**). Tras una corta travesía remera llegaron a la otra orilla y desembarcaron (**CLIMB OUT**).

Poco sabían ambos del peligro que les avecinaba. El bosque de la otra orilla (**E**) estaba plagado de arañas que habían cubierto los árboles de gruesa telas de araña que impedían el paso. Bilbo se abrió paso gracias a la brillante espada que portaba (**SMASH WEB, NE, SMASH WEB, N**).

La agobiante travesía por el bosque de las arañas les condujo hasta un apacible claro. Una puerta mágica se divisaba al otro extremo, suspendida del aire. Ninguno de los esfuerzos que hicieron ambos amigos consiguió que se abriese lo más mínimo. Bilbo, enredando en sus bolsillos, encontró algo metálico que, fortuitamente, fue a parar a uno de sus regordetes dedos (**WEAR RING**). Era el anillo que había encontrado en las cuevas de los goblins. Dos cosas pasaron que dejaron sorprendido al hobbit. La primera es que Thorin lanzó una exclamación y, aunque parecía que miraba en la dirección de Bilbo llamaba a gritos al hobbit como si este hubiera desaparecido. La segunda es que algo alertó a Bilbo de la cercanía de elfos (**EXAMINE DOOR**). Bilbo esperó



unos instantes (**WAIT**) sin anticiparse a los acontecimientos que habrían de desarrollarse. Un par de elfos aparecieron por la puerta, la cual se abrió sin ninguna dificultad, y pasaron de largo, haciendo caso omiso del sorprendido Thorin. Rápidamente Bilbo atravesó la puerta ya que algo le decía que iba a cerrarse a más no tardar (**NE**). Con las prisas tropezó y el anillo se deslizó fuera de su dedo. Thorin volvió a lanzar una exclamación a Bilbo al verle reaparecer de la nada. Bilbo agarró a Thorin y ambos cruzaron la puerta mágica que se cerró a sus espaldas dejándoles en el interior de lo que parecía ser una bodega.

Caminaron en la oscuridad (**S**) hasta llegar a un almacén lleno de barriles. Había alguien más que arrastraba los barriles dejándolos caer por una trampilla a la corriente del río que pasaba por debajo. Bilbo deslizó el anillo (o el anillo se deslizó por voluntad propia, no estaba muy seguro) en su dedo (**WEAR RING**). Thorin intentó esconderse pero era tarde. El individuo de los barriles agarró a Thorin y lo lanzó dentro de lo que debía ser una celda que cerró con llave. Bilbo, mientras el individuo seguía apilando barriles, se deslizó a su espalda y cogió la llave de su cinturón (**GET RED KEY**). Como quiera que el tipo parecía distraído en el fondo del almacén con un par de barriles que pesaban más de la cuenta, decidió dirigirse a la celda donde estaba Thorin y liberarle (**UNLOCK RED DOOR WITH RED KEY, OPEN DOOR**). Como quiera que el de los barriles acababa de lanzar uno al rio de abajo Bilbo trazó mentalmente un plan. Le dijo a Thorin que saltara al barril que todavía se mecía cerca de la trampilla abierta (**SAY TO THORIN "JUMP ONTO BARREL"**). El esperó a que el de los barriles trajera otro y lo arrojara por la trampilla, momento en el cual saltó al interior del barril (**JUMP ONTO BARREL**).



La travesía fue corta y al final llegaron a la Ciudad del Lago, destino último de los barriles (**E**). En la ciudad se encontraron con el bardo, que se ofreció a acompañarles en su viaje hacia la guarida del viejo Smaug (**PICK UP BARD**). Thorin influyó mucho en la decisión de este ya que se puso a cantar sobre oro y sobre las riquezas que acumulaba el dragón.

La jornada fue larga y pasaron por tierras desoladas por el viejo lanzador de fuego (W, N, U, U, N, NW, N). Llegaron a la falda de la montaña Solitaria, lugar donde Smaug tenía su morada. Tal como rezaban las viejas historias esperaron a que los últimos rayos de sol incidieran sobre la ladera (WAIT). En el momento que el primero de los rayos se proyectó contra la roca, una puerta secreta, hasta entonces oculta, quedó a la vista. Thorin se encargó de abrirla (SAY TO THORIN "UNLOCK DOOR WITH CURIOUS KEY"). El bardo y Thorin se quedaron fuera esperando (DROP BARD) mientras Bilbo se colocaba el anillo e iba en busca del ansiado tesoro con la esperanza de cogerlo y salir de allí antes de que Smaug se diese cuenta (WEAR RING, E, E). La cámara del tesoro era descomunal, el dragón descansaba dentro, lanzando volutas de humo por las fosas nasales. Bilbo se acercó sigilosamente a un montón de monedas y cogió las que pudo (GET TREASURE). Como si notara la presencia de Bilbo, el dragón abrió los ojos y olisqueó al aire. Bilbo, paralizado por el terror se quedó inmóvil, Smaug había notado su presencia, le había olido y aunque no le veía eso no iba a impedir que el Hobbit fuese achicharrado en unos instantes. Presa del pánico Bilbo hechó a correr hacia la salida (E, W) con el furioso dragón siguiéndole de cerca. Thorin y el bardo, que esperaban fuera, oyeron los gritos de aviso de Bilbo y los rugidos del dragón. Bilbo le gritó al bardo para que disparara contra el dragón que ganaba terreno por segundos (SAY TO BARD "SHOOT DRAGON"). Un certero disparo y Smaug cayó abatido por la flecha del bardo.

Había sido toda una hazaña. Nuestros exhaustos amigos ya podía emprender la vuelta a casa. Tenían el tesoro y se habían deshecho de una de las bestias más malignas desde hacía siglos. Se despidieron del bardo y emprendieron el largo camino que les esperaba (S, S, S, D, S, S, S, WEAR RING, W, WAIT, WAIT, W, WAIT, WAIT, W, N, SW, W, W, W, W, SW, W).

El agujero donde vivía Bilbo no era grande pero sí muy cómodo. Allí estaba todo tal y como lo había dejado hace días, cuando un grupo de enanos con Gandalf a la cabeza le habían sacado de su cómoda vida y le habían embarcado en la mayor aventura de todas. Bilbo hizo lo último que le quedaba por hacer, guardó su parte del tesoro (OPEN CHEST, PUT TREASURE IN CHEST) y se sentó junto al fuego a descansar."

Xzira

El Claro de los Trolls

Te encuentras en un pequeño claro, semioculto entre los árboles.

Grim Fandango

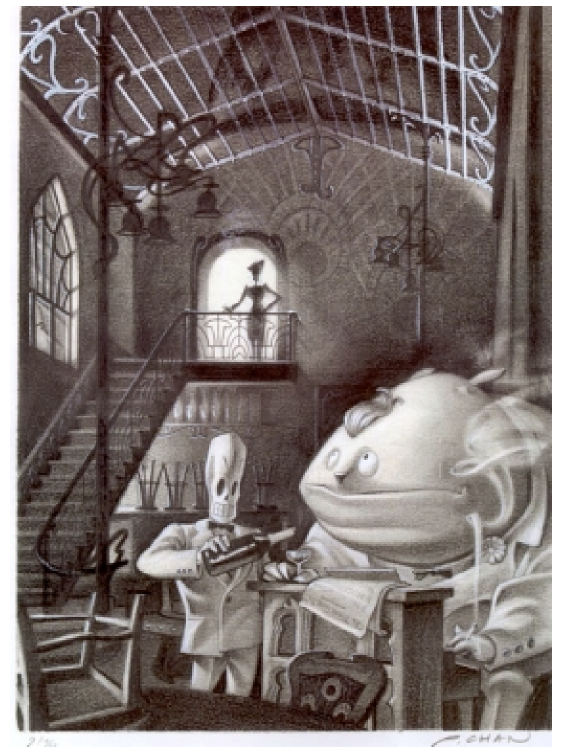
- ¿Hay vida tras la muerte? En LucasArts, ¡sí! -

Historia...

En este caso tomaremos el papel de Manny Calavera (en Español, en el original :), un agente de viajes muy especial, ya que su trabajo consiste en buscar el mejor medio de transporte para sus clientes, las almas que han de recorrer la tierra de los muertos durante tres años.

Por desgracia Manny no esta pasando sus mejores momentos y va a necesitar de toda nuestra ayuda cuando envié a una de sus clientas erróneamente a través del camino más peligroso posible. Y hasta aquí puedo contar, ya que si continuase eliminaría gran parte de la diversión de esta aventura.

El guión es, en mi opinión, el más solido y trabajado de todas las aventuras de Lucas,



mezclando elementos del folklore mejicano y las películas de cine negro de los años cincuenta, con referencias que harán las delicias de cualquiera mínimamente cinéfilo. La historia se desarrolla en tres partes (aumentando la dificultad progresivamente en cada una de ellas), una por cada año que llevará a nuestro protagonista recorrer la tierra de los muertos, a través de los cuales no solo viajaremos sino que viviremos intensamente cada una de las etapas (y es que nadie puede negar a Manny que es un "hombre de recursos").

Pero por suerte nuestro protagonista no está solo en este largo caminar; le acompaña su amigo y conductor Glotis, un demonio elemental de gran

"personalidad" y aun mayor destreza como mecánico. Es bastante inusual la inclusión de un personaje acompañante en las aventuras de Lucas, excepto en el caso del "Indy en busca de Atlantis" (aunque en ese caso, Sofía prácticamente no era más que una pelele proporciona-pistas); y aunque ya todos sabemos del buen hacer de esta compañía con los secundarios ocasionales (quien puede olvidar a Herman Toothrot, o al Tentáculo Verde) que en esta ocasión también lo hacen tan bien como siempre (magnifico Salvador Limones); es con Glotis con quien rizan el rizo dando una lección magistral sobre este tipo de personajes, aprendiendo de los errores que cometieron con Sofía, y creando un personaje con personalidad y cuya relación con nosotros, Manny, evolucionará a lo largo de la historia hasta crear un vinculo que ya no podremos romper.

¿Por último, que sería de una aventura sin un contrincante, un villano? En este caso la plaza esta repartida entre varios personajes, a saber: Domino, el principal contrincante de Manny en la agencia, todo un ganador; Don Copal, jefe de Manny, explotador, malhumorado y sin escrúpulos (como vemos, el típico jefe); y por ultimo, Hector LeMans, un villano a la altura del mismísimo LeChuck (Si, si, no me retracto, a la altura del mismísimo LeChuck).

Graficos...

Grim Fandango es LA PRIMERA AVENTURA 3D DE LA HISTORIA, o al menos eso rezaba la publicidad con la que el juego fue lanzado al mercado. En realidad, lo único realmente 3D son los personajes y algunos elementos del escenario, pero todos los fondos y localidades estaban prerenderizados en 3D desde distintos puntos de vista, por lo que el resultado final es bastante similar al del "Alone in the Dark". Gracias a esto el juego es capaz de funcionar bastante bien en equipos pentium 166, sin aceleración 3D hardware. Un detalle importante



es que con 32 Megas de memoria el juego suele cascar tras avanzar unas pocas pantallas, por lo que es recomendable bajarse un parche que Lucas ha puesto en su pagina web a disposición de los (sufridos) jugadores con este problema.

Sonido...

La banda sonora puede definirse con una sola palabra: Sublime. Los ritmos de jazz y la música suave se deslizan por la historia, ambientando adecuadamente cada uno de sus momentos, y realzándolos magníficamente. El único defecto que tiene es que no es posible comprar un CD aparte con ella para poder escucharla en todo momento.

En cuanto al doblaje, es una grata sorpresa después de la decepción sufrida con el "Curse of Monkey Island". Aparte de un par de frases con alguna entonación inadecuada (algo perfectamente excusable), es muy bueno, especialmente en el caso de Manny y de Glotis, o el de Salvador Limones, proporcionando personalidad a los personajes y atmósfera a las situaciones.

Controles...

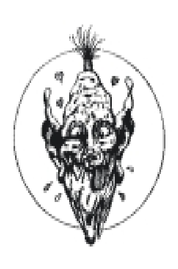
¿Por qué este gran juego no está en el altar que se merece? La razón, creo yo, la podemos encontrar en la decisión de Lucas de hacer desaparecer su famoso sistema de control. Si recordamos en los primeros juegos existía una barra con distintas acciones que combinadas con los elementos de nuestro inventario y/o de la localidad, nos permitían jugar la aventura. Esta barra fue haciéndose cada vez más pequeña, hasta que en "Curse of monkey Island" solo constaba de tres verbos y ni siquiera aparecía en pantalla hasta que pulsabas un botón del ratón. Pero es que en esta ocasión no es que este oculta... simplemente, no esta! ¿Significa eso que Lucas ha abandonado el SCUMM? Nada más lejos de la realidad. En esta ocasión y, por primera vez (y espero que por última), el hecho de no poder seleccionar las acciones en pantalla no significa que no podamos hacerlas. Simplemente significa que se han trasladado hasta nuestro teclado. En esta aventura para mover a Manny usaremos los cursores, para utilizar algún objeto usaremos la tecla "Intro" y para saber que objetos hay en una escena solo tendremos que fijarnos en Manny, que nos lo indicará al volver su mirada sobre todo aquello que sea utilizable. La excusa de Lucas: "La inmersión del jugador en la aventura es así mayor". La realidad: Un experimento medio fallido, que requiere de un periodo de

adaptación antes de sentirte a gusto con la aventura. Especialmente mala ha sido la decisión tomada con el inventario; en vez de tener un panel al que poder acudir a por nuestros objetos, deberemos verlos uno a uno hasta dar con el que queremos usar. Puede que la decisión adoptada sea más "realista" pero la tradicional era, desde luego, mucho más práctica.

Conclusión

Una vez que superas el choque fruto del cambio en los controles, disfrutas de cada momento, de cada situación, de cada personaje... hasta desear que no acabe nunca, y poder quedarte todo el tiempo posible en la tierra de los muertos, tres años y lo que haga falta.

Ambientación	10
Jugabilidad	8
Gráficos	9
Sonido	10
Guión	10
Dificultad	7
Valoración general	9



Roberto Ruisánchez (|Deckard|)

La Torre de Babel

Mucha gente se afana en este impresionante edificio, entre una confusión de lenguajes. Un guía se adelanta entre todos y te saluda en tu idioma: "Permíteme presentarte alguna de nuestras obras, amable viajero".

Historia de Infocom (o... ¿cómo se hizo Zork?). Capítulo 1

Siempre que se habla de los orígenes de alguna de las míticas compañías aventureras que nacieron sobre finales de los 70 o principios de los 80 (y creedme que Infocom lo es, al menos fuera de nuestras fronteras), hay que referirse a la aventura conversacional por excelencia, Adventure de Crowther y Woods.

En 1977 Adventure irrumpió en la ARPAnet y, como consecuencia, llegó al MIT. La reacción de los inquietos chicos del MIT fue la típica de todo el que entró en contacto con aquel juego. Después de invertir un tiempo en resolver el juego, las mentes más inquietas empezaron a pensar en cómo mejorarlo.

Entre estas mentes inquietas nos encontramos con Marc Blank, Bruce Daniels, Dave Lebling y Tim Anderson. Dave escribió (en un lenguaje denominado MUDDLE que posteriormente se pasaría a llamar MDL) un parser de comandos que era tan sofisticado como el de Adventure (es decir, capaz de entender sentencias compuestas por verbo y nombre básicamente), Marc y Tim usaron este parser para escribir el prototipo de un juego de cuatro habitaciones. La verdad es que el juego no era nada del otro mundo por lo que pronto Marc, Bruce y Tim se unieron para intentar escribir un juego en condiciones. Dave estaba un poco apartado del proceso.

Pronto había una serie de mapas y puzzles sobre la mesa. Así nació Zork. Realmente Zork no se llamó "Zork" inicialmente. Zork era el nombre temporal que daban estos muchachos a los programas hasta que estaban listos para ser instalados en el sistema.

Cuando Dave volvió se encontró con un juego más o menos funcional, no tan grande como Adventure pero tenía un ladrón, un cíclope, el troll, el estanque y la presa, la casa, parte del bosque, el glaciar, el laberinto y un montón de cosas más. Los puzzles no eran tan interesantes como los que vendrían más tarder pero por aquella época la gente tenía problemas para imaginar puzzles que incluir en las aventuras, llevó bastante tiempo que la gente aprendiese a implementar puzzles interesantes y, de todas formas, los parser de entonces no soportaban soluciones complicadas.

Zork, al igual que Adventure, sobrevivió gracias a que gente de fuera lo jugaba. En el caso de Adventure esto era posible ya que estaba escrito en FORTRAN y podía ejecutarse en prácticamente cualquier máquina. Zork estaba escrito en MUDDLE que sólo corría en los PDP-10 de DEC así que sus jugadores eran gente que se conectaba a las máquinas del MIT a través de la red (la seguridad a nivel de redes de aquella época brillaba por su ausencia). Los primeros jugadores de Zork comprendían a gente que iba desde John McCarthy, el inventor del LISP, hasta un chaval de doce años de Virginia del Norte.

Esta gente normalmente conseguía "fisgar" la consola de alguien jugando a Zork y, al ver que era un juego del estilo de Adventure, pasaba poco tiempo hasta que conseguía averiguar como ejecutarlo. Durante mucho tiempo el hechizo mágico fue ":MARC;ZORK". Gente que nunca había oído hablar sobre los PDP-10 de alguna forma se enteraron de que si eran capaces de llegar hasta

algo llamado "host 70" en la ARPAnet, se conectaban y tecleaban las palabras mágicas podían jugar al juego.

Aunque en 1977 Zork era mucho más primitivo que el Zork I actual, tenía ya alguno de sus componentes como la familia Flathead, representada por el personaje de Lord Dimwit Flathead the Excesive, gobernante del Great Underground Empire (GUE), y que la moneda oficial era el zorkmid. Los grues, seres que aparecían al caminar por las zonas oscuras del mapeado y acababan con el pobre aventurero, fueron inventados por Bruce en sustitución a los, igualmente funcionales pero menos interesantes, agujeros sin fondo.

El mayor cambio en el juego, hecho en Junio de 1977, fue la introducción del río, diseñado e implementado por Marc. Para ello Marc introdujo un nuevo concepto en el juego, los vehículos. Los vehículos eran objetos que, efectivamente, eran habitaciones móviles. Esto produjo problemas dentro de la estructura del mundo ya que, en principio, el bote sólo estaba codificado para usarse en el río pero nada impedía a los jugadores llevarse el bote deshinchado al estanque e intentar navegar hacia el otro extremo. Es más, los propios autores se quedaron sorprendidos debido al uso que le daban algunos avezados jugadores al bote como bolsa: el bote hinchado contenía los objetos pero el jugador lo transportaba deshinchado. Zork empezaba a cobrar vida propia y eso que apenas contaba con un mes desde su nacimiento.

Bruce seguía, al margen del desarrollo, inventando nuevos puzzles para el juego. Cuando le solicitaron una sección que resultase complicada él diseñó la mina de carbón. El diseño original era mucho más difícil de lo que se implementó al final. En general la mina de carbón es un ejemplo claro de como complicar las cosas sin más que hacerlas tediosas.

El volcán fue la segunda sección que Marc diseñó que requería el uso de vehículos, aunque quizá se más conocida por los bonitos retratos de Lord Dimwit Flathead the Excessive que decoraban la moneda y el sello que había en esta parte. El diseño del volcán requería el uso de otro concepto más, el paso del tiempo. Marc tuvo que añadir un proceso "demonio" que procesaba colas de eventos que ocurrían un número de turnos más tarde de que se disparase el proceso. Esto permitía, además del movimiento de vehículos, manejar el

fusible, la linterna que se apaga y los misteriosos gnomos que aparecían de vez en cuando.

Después de todos estos cambios no se añadieron más partes al juego durante meses. Esto permitió portar el juego a otras máquinas ampliando el número de jugadores de Zork.

En las primeras versiones los jugadores no podían guardar el estado del juego debido a que el método usado requería cientos de miles de bytes por estado guardado, esto era algo excesivo para los sistemas de tiempo compartido de la época. Marc creó una nueva forma de guardar el estado del juego que requería mucho menos espacio, pero tenía la desventaja de que si se añadían nuevas habitaciones u objetos al juego, las posiciones guardadas con versiones más viejas no eran válidas. De cualquier forma esto era mejor que nada.

En otoño de 1977 el juego sufrió dos nuevas modificaciones. La sección de Alicia en el país de las maravillas que incorporaba un cubo mágico y un robot. El robot fue el primer personaje al que el jugador podía transmitirle órdenes de la forma: "ROBOT, TAKE THE CAKE".

En la misma época se añadió la primera versión del sistema de combate. Dave, que era un antiguo jugador de Dungeon&Dragons, creó un sistema de lucha al estilo D&D con diferentes puntos de daño según el arma, heridas, inconsciencia y muerte. Cada criatura tenía su propio juego de mensajes de tal forma que la lucha con el ladrón era muy diferente que la lucha con el troll y su hacha.

Inicialmente, Zork sólo podía obtenerse como ejecutable, nadie (excepto los desarrolladores) tenían acceso al código. Estos mantenían los fuentes encriptados y había "parcheado" el sistema para proteger el directorio donde mantenían los fuentes. Como suele ocurrir en estos casos, un hacker consiguió acceder a los fuentes, saltándose los sistemas de seguridad que los desarrolladores de Zork habían puesto.

Como consecuencia de esto, un fanático de Zork decidió convertirlo a FORTRAN. Los desarrolladores habían asumido que esto resultaba imposible ya que MUDDLE era muy diferente a FORTRAN, mucho más complicado y habían usado todas sus características para implementar Zork. Se equivocaron. En 1978 ya había una versión en FORTRAN, inicialmente para los PDP-11, que podía ejecutarse en la mayoría de las máquinas.

La siguiente sección que se añadió al juego debía ser la última: después de que el jugador había conseguido todos los puntos posibles debía jugar el final del juego, diseñado por Dave. Esta posteriormente se convertiría en la sección del Zork II con el Dungeon Master.

Hacia finales del verano de 1978 se añadieron las sección del Royal Zork Puzzle Museum y de las Palantirs. Al igual que Adventure, como tributo, se añadió el último punto de la aventura, aquel que se conseguía por dejar un objeto en particular en una habitación sin una razón especial.

En Febrero de 1979 se añadió el último puzzle al juego y los desarrolladores se dedicaron al corregir "bugs" la juego durante casi dos años más. La última actualización creada para los mainframes está fechada den Enero de 1981. No se añadieron más puzzles por varias causas: los desarrolladores no tenían el tiempo ni las ganas y no había más espacio disponible ya que estaban limitados a no superar 1 megabyte de memoria y ya lo habían usado por completo.

Hacia Junio de 1979, diez socios formarán la compañía Infocom: Tim Anderson, Joel Berez, Marc Blank, Mike Broos, Scott Cutler, Stu Galley, Dave Lebling, J. C. R. Licklider, Chris Reeve, Al Vezza. En principio se iban a dedicar a programar y distribuir software de gestión pero, por una serie de azares del destino, se decide que su primer producto comercial será Zork.

Referencias

Página "oficial" de Infocom: http://www.csd.uwo.ca/Infocom
Web de Activision: http://www.activision.com

Javier San José (JSJ)

El Pantano

Una voz susurra entre la niebla del pantano: "No puedes hacer eso."

Lola huele bien y adora el marisco

Esta es la primera aventura que he jugado de las presentadas al concurso de aventuras del 98. Reconozco que la elegí en primer lugar solamente por el nombre pero la verdad es que me ha sorprendido gratamente: el lenguaje utilizado es muy gracioso (los textos se salen) y además es bastante sencillo tanto jugar cómo avanzar...

Sin embargo, a mi juicio, tiene algunos pequeños fallos:

- es la típica aventura en la que hay que escribir la orden exactamente cómo la ha pensado el autor y en un orden muy concreto con lo que su nivel de dificultad crece bastante.

- tiene varios errores de programación pero a mi me han servido para darme cuenta de mis propios errores en la aventura breve que estoy realizando. Un ejemplo; si dejas la toalla en cualquier localidad que no sea la de Lola y luego la coges comprenderás a lo que me refiero (¡banderas al poder!).

- la forma de hablar con los PSI no me gusta. Tener que escribir siempre antes 'decir PSI' cómo una orden separada de lo que se quiere decir me parece una clara limitación del parser. Por suerte te acostumbras (¿o no?).

- el final tiene algo incoherente con el desarrollo de la aventura. A ver si alguien nota el detalle al que me refiero ;-). Supongo que el autor escribió el final antes de tener completamente decidido el desarrollo de la aventura y por ello se colaría ese fallito.

- y bueno, si tuviera gráficos y sonidos sería la reostia :-)

Por lo demás solo puedo decir que me gustó mucho y que no pude dejarla hasta que conseguí tirarme a la Lola. De hecho, me estuve documentando largo y tendido (literalmente) sobre el tema con varias de mis novias. Espero que las otras aventuras presentadas sean también de un nivel alto, aunque espero poder comentároslo según las vaya probando (o acabando, seamos optimistas).

Finalmente os diré que creo que si no fuera porque esta aventura infringe la regla de no estar publicada antes del 01/06/2000, si se presentase al concurso de aventuras breves sería una seria candidata al mejor PSI con Lola (y no quiero ofender a Xzira, pero es que Lola es menos facilona y más neurótica, cómo una tía real, vaya).

La solución

Te recomiendo que la intentes acabar por tus medios, pero si te quedas atascadísimo, ¡que demonios!, échale un vistacillo que todos hemos pecado alguna vez...

Es una solución lineal (no me gusta que se interpreten las acciones en las soluciones pues condicionan la interpretación de la aventura al jugador). He procurado poner solamente lo necesario para acabarla sin entrar en detalles (te los dejo a ti y no te arrepentirás ya que los textos son 'cojonudos'). Quizá alguno vea raro que use infinitivos pero yo aprendí a jugar así y con los años parece que no he perdido la mala costumbre :-(

Premisas

- En las localidades nudistas has de quitarte el tanga o el vigilante te ostia.

- En las localidades no nudistas has de llevar puesto el tanga o el vigilante te ostia.

- Si ofendes a Lola ella pasará de ti, pero te perdonará si escribes estas dos órdenes (por ejemplo):

> *decir lola*
> *soy gilipollas*

- Si la ofendes dos veces estás jodido.

Hala, ya está aquí la solución

> *quitar tanga*
> *decir lola*
> *eres inteligente*
> *decir lola*
> *apetece polvo*
> *desnudar lola*
> *decir lola*
> *apetece polvo*
> *besar boca*
> *norte*
> *norte*
> *esperar*
> *sur*
> *este*
> *besar boca*
> *besar tetas*

> *magrear tetas*
> *magrear chocho*
> *besar chocho*
> *decir lola*
> *dame condon*
> *norte*
> *poner tanga*
> *sur*
> *derrumbar castillo*
> *examinar cosa*
> *coger condon (puagggg!!!)*
> *norte*
> *limpia condon (no acepta el infinitivo)*
> *quitar tanga*
> *este*
> *poner condon*
> *follar lola*
> *decir lola*
> *dame condon*
> *follar lola*
> *decir lola*
> *chupa polla*
> *follar lola*

¡¡¡Oe, oe, oe, oe!!!

Un par de curiosidades

La "frase"; me salió en un momento de completa desesperación pero es ideal para comprobar la potencia de cualquier parser y además es un buen exponente de la versatilidad de la lengua castellana:

> *hazte una paja y que le den por culo a la calientapollas esta*

Véngate (un poco) del vigilante de la playa. En la pantalla de la playa abarrotada:

> *golpear vigilante*

Francisco Prieto

El Castillo del Viejo Archivero

Noche invernal en los Cárpatos. Una de esas noches en las que sólo la lívida luna se atreve a asomarse tímidamente entre las gélidas nubes para responder a los pavorosos aullidos de los semicongelados lobos esteparios. Dentro del castillo del Archivero, el fuego chisporrotea alegre con la necesaria fuerza para entibiar los siempre congelados huesecitos del dueño.

Los huesudos dedos rebuscan afanosamente en el viejo baúl. Pronto topan con algo que los legañosos ojillos miran con regocijo: las fotos de **Madridaventura 1999**...



Manowar y JSJ (¿qué estarán tramando?).



El grupo se toma un merecido descanso.



Cárdenas ahoga sus penas con cerveza enana (que tiene cierta similitud con la leche).

Las aventureras también tuvieron representación.



El Interior de la Espiral

La rampa te conduce hasta un majestuoso claro. Ante ti se eleva la Espiral, pulida, brillante y sobre todo impresionante. En su interior...

Programando aventuras conversacionales multiplataforma

Quizá poca gente hoy en día se plantea la realización de una aventura sin un parser, es decir, directamente en C, Pascal, C++ o incluso Basic.

Sin embargo, los conocimientos sobre la realización de programas multiplataforma (pues lo explicado aquí puede generalizarse a cualquier programa) siempre serán necesarios para programar dichos parsers o bien librerías de parsing.

Básicamente, es necesario clarificar conceptos a dos niveles: primero, el diseño de una arquitectura correcta para la aventura multiplataforma, y segundo, la elección de un lenguaje adecuado, multiplataforma a su vez.

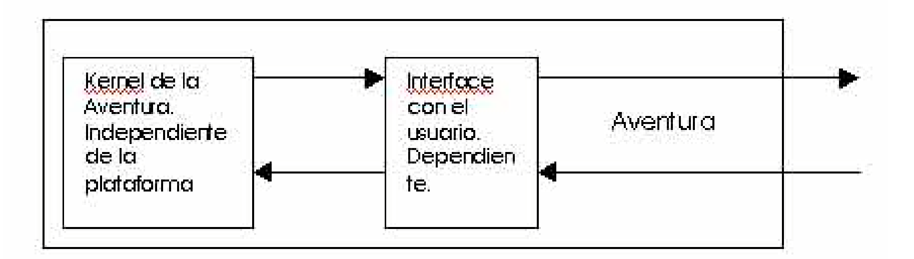
La arquitectura

En primer lugar, es necesario planificar correctamente la aventura, desde un nivel al que quizás no estemos acostumbrados: lo normal es separar la funcionalidad de la aventura en módulos, o en clases, aplicando la técnica "divide y vencerás". Sin embargo, necesitaremos en este caso un paso previo: separar nuestro programa en dos grandes bloques: el encargado de la e/s (entrada/salida), es decir, el interface con el usuario, y el encargado realmente del *parsing*, del control de objetos, de localidades, y de PSI's (Personajes pSeudo-Inteligentes). Puede parecer que la primera va a ser muchísimo más pequeña que la segunda, sin embargo descubriremos con sorpresa que ambas, finalmente, serán de un tamaño similar. Pensemos en el caso del *parser* Inform: la entrada para la generación de aventuras es un fichero de texto que es el fuente de la misma. Podemos crear (nadie nos lo impide) un editor avanzado visual que finalmente, tras definir una aventura con unos cuantos golpes de ratón (es un decir), genere código fuente Inform. Ahí lo tenemos: las dos partes bien diferenciadas: la que depende de la plataforma (*interface* con el usuario) y la que no lo

es: el *kernel* real, el parser Inform, que sólo trabaja contra un fichero de código fuente.

Puede parecer que la elección de un lenguaje multiplataforma puede evitar este problema si utilizamos las características básicas de e/s, como pueden ser las funciones `write`() en Pascal, `printf`() en C o el objeto `cout` en C++. Sin embargo, esto tiene dos inconvenientes graves: el primero es que provocaremos, incluso sin proponérnoslo una interdependencia enorme entre la consola y la aplicación; y finalmente, y como consecuencia de la primera el juego será inadaptable a nuevos entornos como por ejemplo Windows(Win32) o XWindows.

Por tanto, es necesario conformar nuestra aplicación de la siguiente forma:



El lenguaje ANSI/ISO C++

En esta sección propongo la utilización de un lenguaje que es aceptado en muchas plataformas, como el C++. La otra condición necesaria es que esos compiladores realmente acepten el mismo lenguaje, es decir que exista un estándar para él, como existe para el C o el Pascal.

La primera pregunta que puede plantearse es: ¿qué significa ANSI/ISO C++?, ¿qué implica para mí como programador de C++?. ANSI e ISO no son más que los nombres de dos comités internacionales (el primero ligado a Norteamérica, y el segundo internacional) de estándares. Cuando un lenguaje alcanza cierto prestigio, se plantea la necesidad de crear un estándar para él, de forma que alguien independiente a los intereses de mercado dicte el camino que debe seguir la evolución del mismo. Basic, Pascal, y C son lenguajes que poseen sus propios estándares, lo que permitió en su momento que diversos compiladores fuesen creados para ellos, por distintos fabricantes y para distintas plataformas, soportando el mismo lenguaje. Un ejemplo muy sencillo de comprender es el de C. Existen compiladores de C para UNIX, Linux, Win32..., pero todos soportan el mismo lenguaje, de forma que

para muchas aplicaciones Linux se distribuye el código fuente, quedándole al usuario la tarea de compilarlo en su compilador de C para poder ejecutarlo en su máquina.

La estandarización siempre es buena, pues impide que un lenguaje crezca sin control o medida, de una forma racional, y guardando siempre, en la medida de lo posible, la compatibilidad hacia atrás. En el caso de C++, el estándar se aprobó a finales de 1997, entrando en vigor a mediados de 1998. De todas formas, todos los fabricantes de compiladores han seguido muy de cerca la evolución de éste, así que ya a principios de 1998 la mayoría de los compiladores ya lo cumplían: El compilador que incorpora Borland en sus productos Borland C++ y Borland C++ Builder es estándar, el g++ de Linux también lo es, así como el MS Visual C++ 6.0 (con algunos matices). También es posible obtener un compilador gratuito de C++ basado en el g++ de GNU, llamado *dgpp*, en Red Iris (http://www.rediris.es), por ejemplo.

Creando la aventura

Una vez con estas dos bases asentadas, podemos comenzar a hablar de la construcción de la aventura., en cada uno de sus dos partes principales, como se ha dicho.

El Kernel

El *kernel*, núcleo, será le programa principal de la aventura, y a su alrededor dispondremos un *Shell*, la interface de usuario, que será la comunicación con el usuario, de la misma forma en que trabajan, por ejemplo, Linux y Xwindow.

La primera consecuencia lógica que podemos deducir de esto, es que el kernel no va a tener comunicación con el exterior. Es decir, no podemos hacer que el programa de la aventura en sí imprima en pantalla "*no puedes hacer eso*", o "*las runas mágicas tienen un fulgor especial*", o incluso "*Error: posición no disponible en lista de objetos*".

Vamos a necesitar dos formas principales de comunicación, en la que una se corresponde con el usuario, y la segunda con el programador (de forma que el usuario no debería nunca recibir un mensaje de la segunda).

Mensajes de usuario
1. Captación de la linea de órdenes
2. Muestra de mensajes de juego (examinar, descripción de localidad … etc).

Mensajes de sistema
1. Mensajes de error
2. Mensajes de control

Centrándonos en los mensajes de sistema, deberemos sustituir éstos por, bien códigos de error, o bien, amoldándonos en las posibilidades de C++, excepciones. En cualquier caso, ambas exigen un diseño correcto del *kernel*, de forma que el programador siempre pueda saber si la llamada al *kernel*, se ha solucionado de forma satisfactoria o ha ocurrido un error. Una técnica común para solucionar este problema es la de códigos de error, de la misma forma que un programador de C o C++ puede averiguar si ha ocurrido un error en la última llamada. Los códigos de error se suelen implementar haciendo que la llamada devuelva un valor que no está dentro de los valores posibles de retorno, debiendo entonces el programador dirigirse a una variable global para conocer exactamente la naturaleza del mismo.

Ejemplo 1

```
// después de obtener la frase del jugador
if (ob = objetos->busca_nombre(sentencia.objeto)!=NULL)
{
examinar(ob);
}
else visualiza(vectmsg[errno]);
…
const objeto* objetos::busca_nombre(const std::string
&x) const
{
for (unsigned int n=0; n<num_objetos;n++)
if (vect_objetos[n]->dev_nombre()==x)
{
errno = 0;
return &vect_objetos[n];
} errno = errOBJNOEXISTE;
return NULL;
}
```

Obsérvese en el ejemplo 1, como se busca el objeto que el jugador ha metido en la línea de comando, en la clase general de objetos. Si no se encuentra, es decir, si ese objeto no existe en la aventura, se dispone la variable general `errno` para que indique el error de "objeto no existe", y, al retornar, se visualiza el mensaje correspondiente en la pantalla, mediante la función `visualiza()`, que veremos más adelante.

En el siguiente ejemplo, se realiza algo similar mediante excepciones. Quizá el lector encuentre el ejemplo muy complicado: esto se ha hecho así a propósito para mostrar las capacidades de las excepciones. Al comienzo, definimos una serie de clases vacías que simplemente nos permitirán detectar el tipo de excepción a tratar, de una

forma jerárquica. Metemos la llamada a la función en una sentencia `try…catch`, lo que nos permitirá recibir los errores que la función nos lance con la sentencia `throw`. Cómo `Eobjnoexiste` y `Eobjnollevado` derivan de la clase `Eobj`, se gestionan en el `catch (Eobj&)`, y decidimos después cuál de los dos errores es basándonos en si podemos convertir con `dynamic_cast`, el error a uno de ellos `Eobjnoexiste`, o `Eobjnollevado`.

Ejemplo 2

```
class Excp {};
class Eobj : public Excp {};
class Eobjnollevado: public Eobj {};
class Eobjnoexiste: public Eobj {};
…
// después de obtener la frase del jugador
if (ob = objetos->busca_nombre(sentencia.objeto)!=NULL)
{
try {
examinar(ob);
} catch (Eobj &x)
{
if (dynamic_cast<Eobjnollevado &>(x)!=NULL)
visualiza(msgOBJNOLLEVADO);
else if (dynamic_cast<Eobjnoexiste &>(x)!=NULL)
visualiza(msgOBJNOEXISTE);
}
catch (…)
{
visualiza(msgERRORINESPERADO);
}
}
```

```
else visualiza(vectmsg[msgOBJNOEXISTE]);
…
void parser::examinar(objeto *ob)
{
if (ob!=NULL)
if (ob->es_llevado())
{
visualiza(ob->descr());
}
else throw (Eobjnollevado Ex);
} else throw (Eobjnoexiste Ey);
}
```

Repito que puede parecer que este ejemplo es muy complejo, de hecho no es necesario obrar así para cualquier posibilidad de error. La razón es que este ejemplo ilustra varias capacidades del manejo de respuestas con excepciones.

El segundo tipo de posible interacción es la del usuario con el programa, necesitando tomar la frase de acción y precisando además mostrar sus resultados en pantalla (o donde sea).

Podemos observar en los dos ejemplos como se llama a la función `visualiza()`, que es la que gestiona la salida de datos. La cuestión es que esta función será dependiente del interface de la máquina, mientras que necesitamos utilizarla en el módulo del parser. El prototipo de las funciones de usuario podría ser, en el módulo del parser:

```
extern void visualiza(const std::string &);
extern bool pregunta(const std::string &);
```

No he incluido deliberadamente la función de entrada del usuario, puesto que ésta no suele necesitar ser llamada desde el párser. Si se necesita pedir confirmación al usuario, tenemos `pregunta()`, que devuelve verdadero o falso según el usuario responda sí o no. En todo caso, estas funciones son orientativas: quizá los autores encuentren que necesitan alguna más.

Lo que le indicamos con `extern` al compilador es que vamos a utilizar estas funciones, pero que van a estar definidas en otro módulo, es decir, el de *interface*, que es donde estará el cuerpo de esta función. Estas funciones son en realidad funciones de enganche, que comunicarán los objetos encargados de *interface*, con los del *parser*. Por supuesto, puede haber distintas formas de solucionar esto, pero en el ejemplo 3 se muestran unas funciones que enganchan un *kernel multiplataforma* con los objetos de la *interface* de usuario de C++ Builder.

Ejemplo 3

```
Class Tconsola: public Form {
…
Tconsola::operator<<(const std::string &x)
{
Memo1->Lines->Append(x);
}
…
}
inline Tconsola *localiza_consola(void)
{
return static_cast<Tconsola*>(Application->MainForm);
}
void visualiza(const std::string &x)
{
*(localiza_consola) << x;
}
```

`MainForm` en la VCL devuelve el Form principal de la aplicación de Builder. Se busca este Form en la función `localiza_consola()`. La razón de esta función es que es posible que en una futura evolución de la aplicación la consola sea más difícil de localizar que en este caso. De todas formas, al definir la función como `inline`, no perdemos eficiencia en este proceso. Por último, el `Form` tiene sobrecargado el operador << para mandar la cadena que se le pase al `Tmemo` que reside en el mismo.

Conclusiones

Se han presentado, a lo largo de este artículo, una serie de directrices generales a observar para poder realizar aventuras multiplataforma, en la que su principal característica es que precisa de un diseño adecuado de arquitectura así como de análisis y diseño habituales. La segunda característica es que ésta arquitectura se sustentará sobre dos pilares fundamentales: el módulo de *interface*, y el de *kernel* (parser).

Referencias

Allison, C. (1999). "What's new in standard C++?". *C/C++ Users Journal*. Enero de 1999.
Borland Intl. © (1999). Borland C++ Builder Manual and Reference Guide.

José Baltasar García Pérez-Schofield

La Taberna

La taberna "Fuego de Dragón" sirve de albergue a todos los viajeros que se ven obligados a pernoctar después de una dura jornada de viaje.

Reflexiones de un nostálgico.

Me alegra tener que volver a revisar este artículo para el CAAD, por una sencilla razón: el anterior estaba destinado al que iba a ser último número del fanzine. En dicho artículo escribía sobre mi propia "aventura" hasta encontrar el CAAD y la situación aventurera en nuestro país pero, eso sí, bien repleto de pasajes tristes. Ahora que el CAAD ha logrado salir del estado de coma, podré centrarme mejor en los citados puntos.

Mis proyectos en Internet comenzaron hace poco más de dos años, esto es, cuando fundé, junto con otro amigo, el fanzine Macedonia Magazine. Pocas veces me he sentido tan lleno de emoción como cuando colocamos el primer número en la Red. Durante aquellos tiempos había infinidad de errores, pero sobraba ilusión. Sin embargo, la ilusión tiene un apetito voraz y necesita de motivos, por decirlo de

alguna forma, con que alimentarse y reafirmarse. Si durante aquel primer número de Macedonia no hubiéramos tenido la enorme suerte de recibir el email del director del CAAD, Juanjo Muñoz, el fanzine habría tardado mucho más tiempo en madurar y ofrecer algo serio. Por lo tanto, quiero agradecer abiertamente la ayuda prestada por el director del presente Club a Macedonia en sus inicios. En la actualidad, Macedonia se encuentra capitaneada, entre otras personas, por dos aventureros de pro Daniel Cárdenas y Raúl Álvarez, los cuales no creo que necesiten ningún tipo de presentación en estas páginas.

A partir del momento en que conocí quién era Juanjo, el CAAD, su relación con mi añorada Aventuras AD y, en definitiva, todo el romanticismo que encerraba esta publicación aventurera (gracias a largas horas de "travesía" por la página oficial del Club), no pude sino decirme "¿cómo he podido estar perdiéndome todo esto?". Obviamente, pedí a Juanjo una suscripción inmediata.

Como, supongo, todo lector que tiene el placer de leer un CAAD en papel por primera vez, yo quedé enamorado del formato y de los detalles que sembraban toda la edición que acababa de recibir. Es curioso, y parecerá una tontería, pero no tenía la sensación de poseer un tesoro entre las manos desde que, en mi infancia, alternaba el Spectrum con "libro-juegos" como el de la "Búsqueda del Grial", por poner un ejemplo. Quizás fuese ese el motivo que trajo a mi memoria recuerdos de épocas a las que, si tuviera una máquina para viajar por el tiempo, retornaría sin pensármelo dos veces. Estoy recordando, obviamente, los irrepetibles 80.

Los irrepetibles 80.... qué tiempos, ¿eh?. Estoy seguro de que no habrá muchos lectores, españoles, del CAAD, que no hayan vivido aquella época. Una época que nos "robó" una de las etapas más importantes de nuestra vida: la infancia y/o la juventud. Muchas son las imágenes que a un servidor se le pasan por la mente en el preciso momento en que comienza a recordar aquellos años, sin embargo, existen algunas que se presentan de forma insistente. Personalmente, siempre he considerado imágenes inamovibles las apariciones de Andrés Samudio en la MicroHobby o en la MegaOcio. Sus columnas de opinión, el apartado de "El Viejo Archivero" y aquel en el que, el maestro del género en España, enseñaba los conceptos clave para crear una aventura. Los

cuales, por cierto, siguen tan vigentes como antaño.

Ciertamente te da rabia pensar que ya nada volverá a ser como antes. Pero quizás sea ese el motivo por el cual recordamos aquellos tiempos como algo estupendo. ¡Qué raros, y mal vistos éramos entonces los que dedicábamos nuestro tiempo libre a un ordenador!, ¿eh?. Sí, efectivamente fueron buenos tiempos. Resultará gracioso cuando, en las titulaciones universitarias que haya en el futuro sobre videojuegos, que las habrá, se imparta la historia del soft nacional. Al menos nosotros podremos decir que tuvimos el privilegio de vivir la época de oro y de ser aún socios de este histórico y prestigioso club. Vaya, ¡qué bien suena esto!.

Si el amago de desaparición del CAAD se hubiera consumado, habría supuesto un fuertísimo varapalo para muchos nostálgicos de este país. Con el cierre oficial de la publicación desaparecería uno de los últimos reductos nacidos durante la época de esplendor del soft nacional y, cómo no, de la aventura en España. Pero el aviso no ha sido en vano, pues creo que ha servido para que muchos seguidores, de actitud pasiva, reflexionen sobre cuán importante es su colaboración para llevar adelante un género, un proyecto... una ilusión compartida.

La etapa por la que está pasando la aventura clásica no es buena, pero la posibilidad de desaparición del CAAD ha demostrado que "ahí fuera" hay una legión de aventureros que pueden "dar la vuelta al marcador". Es por esta razón por la que me opongo, como muchos de vosotros, ante cualquier idea pesimista sobre el tema. Pienso que el género de la aventura, en cualquiera de sus dos verdaderas vertientes (conversacional y gráfica), debe de saber innovar y evolucionar sin perder por ello su esencia. Está claro que, lo dicho anteriormente, no es sencillo. Resulta más factible abordar géneros en los que el componente técnico es primordial, esto es, en los que la creación del juego se convierte en algo mucho más mecánico pero rentable económicamente. De nuevo, esta justificación no implica la destrucción de las aventuras. Simplemente nos conduce a una reducción del número de producciones, a una mayor especialización del género y a una necesidad, por parte de las desarrolladoras, de seguir adelante a base de títulos originales e inteligentes.

La responsabilidad de que las aventuras comerciales (y recalco lo de comerciales) vuelvan a ocupar el espacio que se merecen no la tienen los usuarios sino los equipos de desarrollo. Son ellos los que deben de captar la atención de nuevos jugadores y saber, al mismo tiempo, no perder a los que ya son entusiastas del género. Desde este punto de vista, las aventuras no están mas que naciendo y se perfilan como los desarrollos que cuenten con uno de los diseños más desafiantes de cara al futuro. Todos tenemos constancia de lo que ha pasado en el mundo de los juegos de rol por ordenador. Los CRPG sí que han estado a punto de desaparecer desde que en 1994/95 se editara el último producto de la serie Ultima y sin embargo, en la actualidad, es el tercer tipo de producción en demanda. No creo que haga falta citar ejemplos de tal hecho. Por lo tanto, no debemos de caer en el error de pensar que la aventura, fuera y dentro de España, es un género muerto. Y no debemos por una sencilla razón: estaríamos pasando por alto, intencionadamente, los ingentes esfuerzos que se están llevando desde el panorama underground y comercial dentro de nuestro país. Muchos de los responsables de tanto esfuerzo son reputados socios del CAAD y aunque se diera el caso de que todos los fanzines sobre aventuras desaparecieran, estoy seguro de que ellos no dejarían de seguir trabajando en pos de la aventura.

Desde luego, no hay género en la historia de los videojuegos que cuente con seguidores tan apasionados como es este de la aventura. El único capaz de seguir divirtiendo aunque sobre él hayan pasado décadas. La aventura es la única "disciplina" del software de entretenimiento cuya esencia no descansa, necesariamente, sobre unos cimientos dependientes de la tecnología y este hecho convierte a cualquier producción en una experiencia intemporal.

Después de la noticia de que el CAAD salía adelante del importante bache en el que se encontraba (pocos fanzine podrían hacerlo), se abre un nuevo periodo, una nueva época, dentro de la historia del club. ¿Cómo será?. Probablemente servirá para que a mucha gente se le termine por despejar la cabeza en ciertos aspectos y para que los que llevan trabajando desde siempre, reciban nuevas dosis de energía y seguridad. Pero, sobre todo, para volver a unir a los verdaderos seguidores de los "clásicos".

Fernando Rodríguez

El Cruce de Caminos

Llegas a un cruce de caminos. Salidas visibles: norte, sur, este y oeste.

Hacia el NORTE puedes ir a:

InformatE (http://www.geocities.com/TimesSquare/Fortress/9939), página de Inform en castellano.

Visual SINTAC (http://personales.jet.es/jsjlogin/sintac), página del parser visual para Windows.

Macedonia Magazine (http://www.codexmx.com/macedonia), otra publicación con una sección (Spanish Quest) dedicada al mundo aventurero.

AventuraST (http://www.civila.com/hispania/elarchivo/conversacional_principal.HTML), una página muy aventurera.

The Roebuck (http://thepentagon.com/nmsoft), la página de Carlos Sánchez.

Fundación Pro-Imagyna (http://www.geocities.com/TimesSquare/5787), la página aventurera de Daniel Cárdenas.

Página de Melitón (http://www.arrakis.es/~meliton) las páginas de este veterano aventurero.

Página de Jaume Alcazo (http://www.geocities.com/TimesSquare/Hangar/4795).

Página de Vicente Tarín Font (http://ttt.eui.upv.es/~vitafon).

Página de LDAP (http://members.es.tripod.de/gen/some.html).

The Dwarf and the Axe (http://www.geocities.com/TimesSquare/3567/spaxe.html)

La Península Aventurera (http://www.geocities.com/TimesSquare/Stadium/5162).

El camino que va hacia el SUR te conduce a:

Adventureland (http://www.lysator.liu.se/adventure), completa información sobre compañías aventureras de todos los tiempos.

Infocom (http://www.csd.uwo.ca/Infocom), página sobre esta mítica compañía.

The Colossal Cave Adventure (http://people.delphi.com/rickadams/adventure/index.html), conoce los orígenes de todo esto

The IF Library (http://www.iflibrary.org), el archivo aventurero por excelencia.

The Spectrum Adventurer (http://home.virtual-pc.com/isblpx), para los nostálgicos.

XYZZY News (http://www.xyzzynews.com), webzine sobre la interactive fiction inglesa.

Hacia el ESTE puede ir a:

The roguelike games homepage (http://www.win.tue.nl/games/roguelike), todo sobre roguelikes.

Thangorodrim (http://thangorodrim.angband.org), página dedicada a Angband, uno de los roguelikes con más éxito.

Hacia el OESTE queda:

Emulatronia (http://www.emulatronia.com), la mejor página sobre emuladores, en castellano.

Si quieres sugerir algún enlace para su inclusión en esta sección, o detectas alguno erróneo, puedes escribir a jsjlogin@jet.es.

I Concurso de Aventuras Breves en Castellano

El concurso tiene como objetivo fomentar la creación de aventuras, así como explotar las posibilidades del género en terrenos como la interacción con objetos y personajes. Además, servirá para ampliar la base de aventuras con código fuente disponible, como fuente de ejemplos para los distintos parsers y lenguajes de programación.

Este concurso se plantea como una extensión lógica a los concursos de mini-aventuras que se han venido convocando hasta la fecha y que tanto éxito han demostrado tener. Esto no debe considerarse en perjucio de los mismos ya que seguirán convocándose regularmente concursos de mini-aventuras.

Plazos

El plazo de recepción de los trabajos finaliza el **30 de Septiembre de 2000** a las 12 de la noche.

Entrega

Las aventuras deben enviarse al CAAD por correo electrónico, a la dirección concurso@grupo-ronda.com

El envío deberá estar dividido en dos ficheros:

- Un fichero contendrá sólo el ejecutable de la aventura. Si lo desea, el autor puede incluir un LEEME donde explique brevemente el objetivo de la aventura o cualquier cosa necesaria para jugarla. Este fichero será expuesto públicamente en cuanto finalice el período de recepción de trabajos.

- Un segundo fichero contendrá un fichero de texto con la solución de la aventura y opcionalmente, el código fuente del programa. El autor puede incluir aquí cualquier comentario o información adicional sobre el juego, según crea él oportuno. Este fichero será expuesto públicamente cuando finalice el período de votación y con ello el concurso en sí.

La inclusión del código fuente de la aventura es opcional, pero muy aconsejable, ya que las aventuras que participen en este concurso serán ejemplos interesantes del uso de sus respectivos parsers y lenguajes.

Los trabajos presentados deben ser originales y no haber sido distribuidos antes del 1 de Junio de 2000. No se exige, sin embargo, que el juego quede oculto hasta el 30 de Septiembre de 2000. El autor es libre de distribuirlo por su cuenta, a riesgo de mostrar su trabajo antes de tiempo (probablemente quedando en desventaja respecto al resto de participaciones).

El autor es libre de presentar aventuras bajo seudónimo. En caso de presentar más de una aventura podrá firmar cada una con un seudónimo diferente si así lo desea. Ediciones CAAD se compromete a mantener la identidad real del autor en secreto durante toda la duración del concurso. En todo caso el autor debe incluir en el segundo fichero (el que contendrá la solución de la aventura) su nombre real.

Jurado popular

El día 1 de Octubre de 2000 se habilitará una zona en la página del CAAD donde podrán descargarse los trabajos presentados.

Cada votación consistirá en puntuar (en una escala de 0 a 10 con hasta 2 decimales) una de las aventuras participantes en el concurso en los siguientes apartados:

- ORIGINALIDAD

¿Nunca se había visto antes este problema o situación?

¿La historia que narra es original?

¿Nos mete en el papel de un personaje cuyo retrato resulta inédito?

- CALIDAD DEL TEXTO

¿Es interesante la historia que se cuenta?

¿Está bien ambientada?

¿Está bien escrito?

¿Carece de faltas de ortografía?

- INTERACTIVIDAD

¿Ofrece el juego respuesta a la mayoría de comandos lógicos?

¿Pueden manipularse a fondo todos los objetos?

¿Dan los personajes bastantes respuestas?

¿Se contemplan los intentos de probar otras soluciones lógicas a los problemas?

- JUEGO LIMPIO

¿Es un problema lógico sin ser demasiado sencillo?

¿Hay pistas adecuadas si el problema es difícil?

¿Se usa adecuadamente la lógica externa?

- ADECUACION A LOS LIMITES

¿Cumple las limitaciones impuestas?. Véase el apartado Limitaciones.

- VALORACION GLOBAL

Impresión general que ofrece la aventura en su conjunto.

Una misma persona puede votar a una aventura, a varias, o a todas, sin ninguna obligación. Los autores también pueden votar a todas las aventuras en concurso con excepción de las que él mismo haya presentado. Para que una aventura pueda optar a premio, debe haber recibido votaciones de al menos TRES personas.

El período de votación finalizará el 31 de Octubre de 2000. Se podrá ampliar el plazo si hay petición de ello y se observa que el número de votos recibidos no es suficiente.

Se contemplarán así mismo las siguientes nominaciones especiales. Cada votante nominará una aventura dentro de cada una de las siguientes categorías:

- MEJOR PSI

La aventura que presente el mejor personaje no jugador.

- MEJOR PUZZLE

La aventura que presente el mejor puzzle, entendiendo como tal aquel obstáculo de la aventura que necesita de cierta deducción para su resolución.

Votos

Los votos se remitirán por correo electrónico a la siguiente dirección: jsjlogin@jet.es. En el asunto del mensaje debe constar el texto:

[VOTOS] AV. BREVES, Nombre del votante.

Por ejemplo:

`Asunto: [VOTOS] AV. BREVES, Javier San José`

Se sugiere usar el siguiente formato para las votaciones:

NOMBRE DE LA AVENTURA: ORIGINALIDAD CALIDAD TEXTO INTERACTIVIDAD JUEGO LIMPIO ADECUACION VALORACION GLOBAL

MEJOR PSI: NOMBRE DE AVENTURA

MEJOR PUZZLE: NOMBRE DE AVENTURA

Por ejemplo:

AVENTURA ORIGINAL: 5 6 6 8 5 6

MEJOR PSI: LA DIOSA DE COZUMEL

MEJOR PUZZLE: JABATO

En cualquier caso no se rechazará ningún voto que no se ajuste a este formato. Esto se debe considerar como una mera sugerencia para facilitar el recuento de votos.

Licencia

Ediciones CAAD no se reserva ningún derecho sobre los programas presentados al concurso. Estos podrán distribuirse según el método y la licencia de uso que prefiera su autor, sin embargo: todas las aventuras presentadas deberán poder distribuirse y ser jugadas gratuitamente, sin ninguna limitación. Todas las aventuras que incluyan el código fuente (lo cual recordemos que es opcional) deberán poder ser estudiadas, modificadas y reutilizadas libremente, por ejemplo utilizando parte de su código en otras aventuras.

Limitaciones

Se considerará que una aventura es breve si cumple todos estos requisitos:

Incluir un máximo de 6 localidades.

Incluir un máximo de 12 objetos o personajes (o cualquier otra combinación, como 2 personajes y 10 objetos).

Poder acabarse razonablemente en menos de 2 horas, suponiendo que el jugador no queda trabado absurdamente en un atasco :-)

Se sugiere no penalizar una aventura en la votación de este apartado simplemente porque contiene objetos "de escenario", cosas examinables, o repuesta a muchas acciones. Estos aspectos enriquecen el juego sin alargar ni engordar su desarrollo.

Queda a discreción de la persona que vota qué es un objeto y qué no, y si el abuso de cosas como

hechizos, objetos transformables u otros "pseudo-objetos" reduce el carácter breve de la aventura.

Todos estos aspectos deberán quedar reflejados en la puntuación que se otorgue al apartado ADECUACION A LOS LIMITES. Se sugiere puntuar un 10 en este apartado si se cumplen todos los límites disminuyendo proporcionalmente esta puntuación si se sobrepasan (hay más de 6 localidades, 12 objetos o el tiempo de juego excede de 2 horas).

Requisitos

La aventura podrá funcionar en cualquier plataforma, aunque una aventura recibirá más votos si todo el mundo puede jugarla.

La aventura podrá incluir atractivos adicionales, como música, gráficos, tipos de letra... Si bien se advierte que ello no se reflejará en la puntuación de la aventura.

Premio

No existe ningún premio para el concurso, salvo la satisfacción y el reconocimiento de todos los aventureros. Sin embargo habrá un ganador del concurso, elegido a partir de la media de todas las puntuaciones en todos los apartados.

El CAAD admitirá, sin embargo, cualquier donación de premio simbólico que alguien desee ofrecer al ganador, por lo que existe la posibilidad de que esto cambie posteriormente.

Se elegirá un ganador del concurso, así como ganadores por apartados. En general, una vez finalizado el plazo de votación, se escogerán:

- LA MEJOR AVENTURA BREVE
- LA AVENTURA BREVE MÁS ORIGINAL
- LA AVENTURA BREVE MEJOR ESCRITA
- LA AVENTURA BREVE MÁS INTERACTIVA
- LA AVENTURA BREVE MÁS JUGABLE

Nominaciones especiales:

- EL MEJOR PSI
- EL MEJOR PUZZLE

Recuento de votos

El sistema de recuento se realizará como sigue:

1.- Por cada categoría se sumará el total de puntos en dicha categoría y se dividirá entre el número de votos recibidos por la aventura en esa categoría.

2.- Se obtendrá un total que será la media aritmética de la suma de puntos del apartado VALORACION GLOBAL y ADECUACION A LAS REGLAS.

La aventura con más puntos en el total será la ganadora (LA MEJOR AVENTURA BREVE). Se elegirán también las ganadoras en las categorías enumeradas en el apartado Premio.

Para las nominaciones especiales se hará una recuento del número de votos y la aventura más votada en cada nominación, será la ganadora.

Empate

En caso de empate en una categoría se decidirá la aventura ganadora en esa categoría por el número de votos. La aventura con más votos en esa categoría será la ganadora.

Si esto no lleva a un desempate se valorará la puntuación total. La aventura con mayor puntuación total será la ganadora en esa categoría.

Pudiera ocurrir que, a pesar de todo, no se llegue a un desempate. En este caso se otorgará el premio a todas las aventuras empatadas.

En el caso de las nominaciones especiales se procederá de manera similar.

Recordamos que cualquier aventura que no reciba al menos las votaciones de TRES personas, quedará fuera de concurso.